

VISIT...

LANZAROTE
Caliente.COM

5

Digital Relaying

5.1	Sampling	5-2
5.2	Antialiasing Filters.....	5-2
5.3	Sigma-Delta A/D Converters.....	5-2
5.4	Phasors from Samples.....	5-4
5.5	Symmetrical Components	5-5
5.6	Algorithms	5-7
	Parameter Estimation • Least Squares Fitting • DFT • Differential Equations • Kalman Filters • Wavelet Transforms • Neural Networks	

James S. Thorp

Virginia Polytechnic Institute

Digital relaying had its origins in the late 1960s and early 1970s with pioneering papers by Rockefeller (1969), Mann and Morrison (1971), and Poncelet (1972) and an early field experiment (Gilcrest et al., 1972; Rockefeller and Udren, 1972). Because of the cost of the computers in those times, a single high-cost minicomputer was proposed by Rockefeller (1969) to perform multiple relaying calculations in the substation. In addition to having high cost and high power requirements, early minicomputer systems were slow in comparison with modern systems and could only perform simple calculations. The well-founded belief that computers would get smaller, faster, and cheaper combined with expectations of benefits of computer relaying kept the field moving. The third IEEE tutorial on microprocessor protection (Sachdev, 1997) lists more than 1100 publications in the area since 1970. Nearly two thirds of the papers are devoted to developing and comparing algorithms. It is not clear this trend should continue. Issues beyond algorithms should receive more attention in the future.

The expected benefits of microprocessor protection have largely been realized. The ability of a digital relay to perform self-monitoring and checking is a clear advantage over the previous technology. Many relays are called upon to function only a few cycles in a year. A large percentage of major disturbances can be traced to “hidden failures” in relays that were undetected until the relay was exposed to certain system conditions (Tamronglak et al., 1996). The ability of a digital relay to detect a failure within itself and remove itself from service before an incorrect operation occurs is one of the most important advantages of digital protection.

The microprocessor revolution has created a situation in which digital relays are the relays of choice because of economic reasons. The cost of conventional (analog) relays has increased while the hardware cost of the most sophisticated digital relays has decreased dramatically. Even including substantial software costs, digital relays are the economic choice and have the additional advantage of having lower wiring costs. Prior to the introduction of microprocessor-based systems, several panels of space and considerable wiring was required to provide all the functions needed for each zone of transmission line protection. For example, an installation requiring phase distance protection for phase-to-phase and three-phase faults, ground distance, ground-overcurrent, a pilot scheme, breaker failure, and reclosing logic demanded redundant wiring, several hundred watts of power, and a lot of panel space. A single microprocessor system is a single box, with a ten-watt power requirement and with only direct wiring, has replaced the old system.

Modern digital relays can provide SCADA, metering, and oscillographic records. Line relays can also provide fault location information. All of this data can be available by modem or on a WAN. A LAN in the substation connecting the protection modules to a local host is also a possibility. Complex multifunction relays can have an almost bewildering number of settings. Techniques for dealing with setting management are being developed. With improved communication technology, the possibility of involving microprocessor protection in wide-area protection and control is being considered.

5.1 Sampling

The sampling process is essential for microprocessor protection to produce the numbers required by the processing unit to perform calculations and reach relaying decisions. Both 12 and 16 bit A/D converters are in use. The large difference between load and fault current is a driving force behind the need for more precision in the A/D conversion. It is difficult to measure load current accurately while not saturating for fault current with only 12 bits. It should be noted that most protection functions do not require such precise load current measurement. Although there are applications, such as hydro generator protection, where the sampling rate is derived from the actual power system frequency, most relay applications involve sampling at a fixed rate that is a multiple of the *nominal* power system frequency.

5.2 Antialiasing Filters

ANSI/IEEE Standard C37.90, provides the standard for the Surge Withstand Capability (SWC) to be built into protective relay equipment. The standard consists of both an oscillatory and transient test. Typically the surge filter is followed by an antialiasing filter before the A/D converter. Ideally the signal $x(t)$ presented to the A/D converter $x(t)$ is band-limited to some frequency ω_c , i.e., the Fourier transform of $x(t)$ is confined to a low-pass band less than ω_c such as shown in Fig. 5.1. Sampling the low-pass signal at a frequency of ω_s produces a signal with a transform made up of shifted replicas of the low-pass transform as shown in Fig. 5.2. If $\omega_s - \omega_c > \omega_c$, i.e., $\omega_s > 2\omega_c$ as shown, then an ideal low-pass filter applied to $z(t)$ can recover the original signal $x(t)$. The frequency of twice the highest frequency present in the signal to be sampled is the Nyquist sampling rate. If $\omega_s < 2\omega_c$ the sampled signal is said to be “aliased” and the output of the low-pass filter is not the original signal. In some applications the frequency content of the signal is known and the sampling frequency is chosen to avoid aliasing (music CDs), while in digital relaying applications the sampling frequency is specified and the frequency content of the signal is controlled by filtering the signal before sampling to insure its highest frequency is less than half the sampling frequency. The filter used is referred to as an antialiasing filter.

Aliasing also occurs when discrete sequences are sampled or decimated. For example, if a high sampling rate such as 7200 Hz is used to provide data for oscillography, then taking every tenth sample provides data at 720 Hz to be used for relaying. The process of taking every tenth sample (decimation) will produce aliasing unless a digital anti-aliasing filter with a cut-off frequency of 360 Hz is provided.

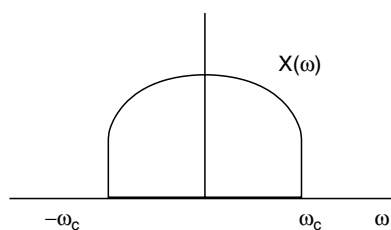


FIGURE 5.1 The Fourier Transform of a band-limited function.

5.3 Sigma-Delta A/D Converters

There is an advantage in sampling at rates many times the Nyquist rate. It is possible to exchange speed of sampling for bits of resolution. So called Sigma-Delta A/D converters are based on one bit sampling at very high rates. Consider a signal $x(t)$ sampled at a high rate $T = 1/f_s$, i.e., $x[n] = x(nT)$ with the difference between the current sample and α times the last sample given by

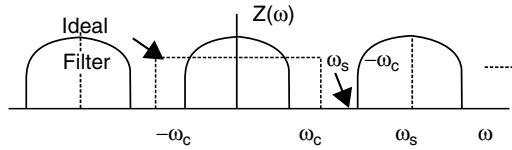


FIGURE 5.2 The Fourier Transform of a sampled version of the signal $x(t)$.

$$d[n] = x[n] - \alpha x[n - 1] \quad (5.1)$$

If $d[n]$ is quantized through a one-bit quantizer with a step size of Δ , then

$$x_q[n] = \alpha x_q[n - 1] + d_q[n] \quad (5.2)$$

The quantization is called delta modulation and is represented in Fig. 5.3. The z^{-1} boxes are unit delays while the one bit quantizer is shown as the box with $d[n]$ as input and $d_q[n]$ as output. The output $x_q[n]$ is a staircase approximation to the signal $x(t)$ with stairs that are spaced at T sec and have height Δ . The delta modulator output has two types of errors: one when the maximum slope Δ/T is too small for rapid changes in the input (shown on Fig. 5.3) and the second, a sort of chattering when the signal $x(t)$ is slowly varying. The feedback loop below the quantizer is a discrete approximation to an integrator with $\alpha = 1$. Values of α less than one correspond to an imperfect integrator. A continuous form of the delta modulator is also shown in Fig. 5.4. The low pass filter (LPF) is needed because of the high frequency content of the staircase. Shifting the integrator from in front of the LPF to before the delta modulator improves both types of error. In addition, the two integrators can be combined.

The modulator can be thought of as a form of voltage follower circuit. Resolution is increased by oversampling to spread the quantization noise over a large bandwidth. It is possible to shape the quantization noise so it is larger at high frequencies and lower near DC. Combining the shaped noise with a very steep cut-off in the digital low pass filter, it is possible to produce a 16-bit result from the one bit comparator. For example, a 16-bit answer at 20 kHz can be obtained with an original sampling frequency of 400 kHz.

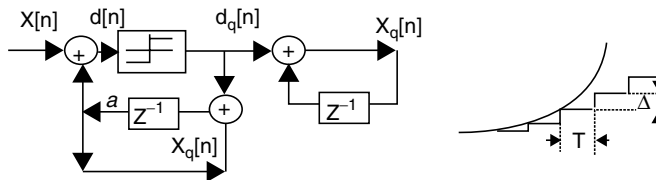


FIGURE 5.3 Delta modulator and error.

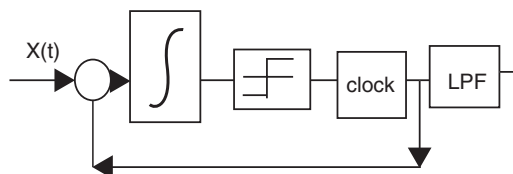


FIGURE 5.4 Signal-Delta modulator.

5.4 Phasors from Samples

A phasor is a complex number used to represent sinusoidal functions of time such as AC voltages and currents. For convenience in calculating the power in AC circuits from phasors, the phasor magnitude is set equal to the rms value of the sinusoidal waveform. A sinusoidal quantity and its phasor representation are shown in Fig. 5.5, and are defined as follows:

Sinusoidal quantity	Phasor	
$y(t) = Y_m \cos(\omega t + \phi)$	$Y = \frac{Y_m}{\sqrt{2}} e^{j\phi}$	(5.3)

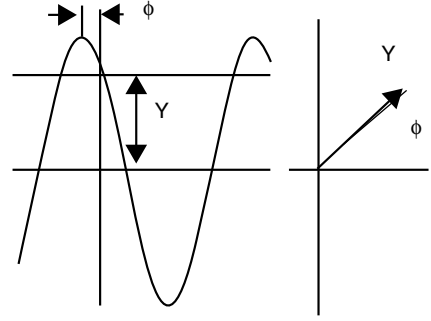


FIGURE 5.5 Phasor representation.

A phasor represents a single frequency sinusoid and is not directly applicable under transient conditions. However, the idea of a phasor can be used in transient conditions by considering that the phasor represents an estimate of the fundamental frequency component of a waveform observed over a finite window. In case of N samples y_k , obtained from the signal $y(t)$ over a period of the waveform:

$$Y = \frac{1}{\sqrt{2}} \frac{2}{N} \sum_{k=1}^N y_k e^{-jk\frac{2\pi}{N}} \quad (5.4)$$

or,

$$Y = \frac{1}{\sqrt{2}} \frac{2}{N} \left\{ \sum_{k=1}^N y_k \cos\left(\frac{k2\pi}{N}\right) - j \sum_{k=1}^N y_k \sin\left(\frac{k2\pi}{N}\right) \right\} \quad (5.5)$$

Using θ for the sampling angle $2\pi/N$, it follows that

$$Y = \frac{1}{\sqrt{2}} \frac{2}{N} (Y_c - jY_s) \quad (5.6)$$

where

$$\begin{aligned} Y_c &= \sum_{k=1}^N y_k \cos(k\theta) \\ Y_s &= \sum_{k=1}^N y_k \sin(k\theta) \end{aligned} \quad (5.7)$$

Note that the input signal $y(t)$ must be band-limited to $N\omega/2$ to avoid aliasing errors. In the presence of white noise, the fundamental frequency component of the Discrete Fourier Transform (DFT) given by Eqs. (5.4)–(5.7) can be shown to be a least-squares estimate of the phasor. If the data window is not a multiple of a half cycle, the least-squares estimate is some other combination of Y_c and Y_s , and is no longer given by Eq. (5.6). Short window (less than one period) phasor computations are of interest in some digital relaying applications. For the present, we will concentrate on data windows that are multiples of a half cycle of the nominal power system frequency.

The data window begins at the instant when sample number 1 is obtained as shown in Fig. 5.5. The sample set y_k is given by

$$y_k = Y_m \cos(k\theta + \phi) \quad (5.8)$$

Substituting for y_k from Eq. (5.8) in Eq. (5.4),

$$Y = \frac{1}{\sqrt{2}} \frac{2}{N} \sum_{k=1}^N Y_m \cos(k\theta + \phi) e^{-jk\theta} \quad (5.9)$$

or

$$Y = \frac{1}{\sqrt{2}} Y_m e^{j\phi} \quad (5.10)$$

which is the familiar expression Eq. (5.3), for the phasor representation of the sinusoid in Eq. (5.3). The instant at which the first data sample is obtained defines the orientation of the phasor in the complex plane. The reference axis for the phasor, i.e., the horizontal axis in Fig. 5.5, is specified by the first sample in the data window.

Equations (5.6)–(5.7) define an algorithm for computing a phasor from an input signal. A recursive form of the algorithm is more useful for real-time measurements. Consider the phasors computed from two adjacent sample sets: $y_k \{k = 1, 2, \dots, N\}$ and, $y'_k \{k = 2, 3, \dots, N + 1\}$, and their corresponding phasors Y^1 and $Y^{2'}$ respectively:

$$Y^1 = \frac{1}{\sqrt{2}} \frac{2}{N} \sum_{k=1}^N y_k e^{-jk\theta} \quad (5.11)$$

$$Y^{2'} = \frac{1}{\sqrt{2}} \frac{2}{N} \sum_{k=1}^N y_{k+1} e^{-jk\theta} \quad (5.12)$$

We may modify Eq. (5.12) to develop a recursive phasor calculation as follows:

$$Y^{2'} = Y^{2'} e^{-j\theta} = Y^1 + \frac{1}{\sqrt{2}} \frac{2}{N} (y_{N+1} - y_1) e^{-j\theta} \quad (5.13)$$

Since the angle of the phasor $Y^{2'}$ is greater than the angle of the phasor Y^1 by the sampling angle θ , the phasor Y^2 has the same angle as the phasor Y^1 . When the input signal is a constant sinusoid, the phasor calculated from Eq. (5.13) is a constant complex number. In general, the phasor Y , corresponding to the data $y_k \{k = r, r + 1, r + 2, \dots, N + r - 1\}$ is recursively modified into Y^{r+1} according to the formula

$$Y^{r+1} = Y^r e^{-j\theta} = Y^r + \frac{1}{\sqrt{2}} \frac{2}{N} (y_{N+r} - y_r) e^{-j\theta} \quad (5.14)$$

The recursive phasor calculation as given by Eq. (5.13) is very efficient. It regenerates the new phasor from the old one and utilizes most of the computations performed for the phasor with the old data window.

5.5 Symmetrical Components

Symmetrical components are linear transformations on voltages and currents of a three phase network. The symmetrical component transformation matrix S transforms the phase quantities, taken here to be voltages E_ϕ , (although they could equally well be currents), into symmetrical components E_S :

$$\mathbf{E}_s = \begin{bmatrix} E_0 \\ E_1 \\ E_2 \end{bmatrix} = \mathbf{S} \mathbf{E}_\phi = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 \\ 1 & \alpha^2 & \alpha \end{bmatrix} \begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix} \quad (5.15)$$

where $(1, \alpha, \alpha^2)$ are the three cube-roots of unity. The symmetrical component transformation matrix $\mathbf{S}$ is a similarity transformation on the impedance matrices of balanced three phase circuits, which diagonalizes these matrices. The symmetrical components, designated by the subscripts (0,1,2) are known as the zero, positive, and negative sequence components of the voltages (or currents). The negative and zero sequence components are of importance in analyzing unbalanced three phase networks. For our present discussion, we will concentrate on the positive sequence component E_1 (or I_1) only. This component measures the balanced, or normal voltages and currents that exist in a power system. Dealing with positive sequence components only allows the use of single-phase circuits to model the three-phase network, and provides a very good approximation for the state of a network in quasi-steady state. All power generators generate positive sequence voltages, and all machines work best when energized by positive sequence currents and voltages. The power system is specifically designed to produce and utilize almost pure positive sequence voltages and currents in the absence of faults or other abnormal imbalances. It follows from Eq. (5.15) that the positive sequence component of the phase quantities is given by

$$Y_1 = \frac{1}{3} (Y_a + \alpha Y_b + \alpha^2 Y_c) \quad (5.16)$$

Or, using the recursive form of the phasors given by Eq. (5.14),

$$Y_1^{r+1} = Y_1^r + \frac{1}{\sqrt{2}} \frac{2}{N} [(x_{a,N+r} - x_{a,r}) e^{-jr\theta} + \alpha (x_{b,N+r} - x_{b,r}) e^{-jr\theta} + \alpha^2 (x_{c,N+r} - x_{c,r}) e^{-jr\theta}] \quad (5.17)$$

Recognizing that for a sampling rate of 12 times per cycle, α and α^2 correspond to $\exp(j4\theta)$ and $\exp(j8\theta)$, respectively, it can be seen from Eq. (5.17) that

$$Y_1^{r+1} = Y_1^r + \frac{1}{\sqrt{2}} \frac{2}{N} [(x_{a,N+r} - x_{a,r}) e^{-jr\theta} + (x_{b,N+r} - x_{b,r}) e^{j(4-r)\theta} + (x_{c,N+r} - x_{c,r}) e^{j(8-r)\theta}] \quad (5.18)$$

With a carefully chosen sampling rate—such as a multiple of three times the nominal power system frequency—very efficient symmetrical component calculations can be performed in real time. Equations similar to (5.18) hold for negative and zero sequence components also. The sequence quantities can be used to compute a distance to the fault that is independent of fault type. Given the ten possible faults in a three-phase system (three line-ground, three phase-phase, three phase-phase-ground, and three phase), early microprocessor systems were taxed to determine the fault type before computing the distance to the fault. Incorrect fault type identification resulted in a delay in relay operation. The symmetrical component relay solved that problem. With advances in microprocessor speed it is now possible to simultaneously compute the distance to all six phase-ground and phase-phase faults in order to solve the fault classification problem.

The positive sequence calculation is still of interest because of the use of synchronized phasor measurements. Phasors, representing voltages and currents at various buses in a power system, define the state of the power system. If several phasors are to be measured, it is essential that they be measured with a common reference. The reference, as mentioned in the previous section, is determined by the instant at which the samples are taken. In order to achieve a common reference for the phasors, it is essential to achieve synchronization of the sampling pulses. The precision with which the time synchronization must be achieved depends upon the uses one wishes to make of the phasor measurements. For example, one use of the phasor measurements is to estimate, or validate, the state of the power

systems so that crucial performance features of the network, such as the power flows in transmission lines could be determined with a degree of confidence. Many other important measures of power system performance, such as contingency evaluation, stability margins, etc., can be expressed in terms of the state of the power system, i.e., the phasors. Accuracy of time synchronization directly translates into the accuracy with which phase angle differences between various phasors can be measured. Phase angles between the ends of transmission lines in a power network may vary between a few degrees, and may approach 180° during particularly violent stability oscillations. Under these circumstances, assuming that one may wish to measure angular differences as little as 1° , one would want the accuracy of measurement to be better than 0.1° . Fortunately, synchronization accuracies of the order of $1\ \mu\text{sec}$ are now achievable from the Global Positioning System (GPS) satellites. One microsecond corresponds to 0.022° for a 60 Hz power system, which more than meets our needs. Real-time phasor measurements have been applied in static state estimation, frequency measurement, and wide area control.

5.6 Algorithms

5.6.1 Parameter Estimation

Most relaying algorithms extract information about the waveform from current and voltage waveforms in order to make relaying decisions. Examples include: current and voltage phasors that can be used to compute impedance, the rms value, the current that can be used in an overcurrent relay, and the harmonic content of a current that can be used to form a restraint in transformer protection. An approach that unifies a number of algorithms is that of parameter estimation. The samples are assumed to be of a current or voltage that has a known form with some unknown parameters. The simplest such signal can be written as

$$y(t) = Y_c \cos \omega_0 t + Y_s \sin \omega_0 t + e(t) \quad (5.19)$$

where ω_0 is the nominal power system frequency, Y_c and Y_s are unknown quantities, and $e(t)$ is an error signal (all the things that are not the fundamental frequency signal in this simple model). It should be noted that in this formulation, we assume that the power system frequency is known. If the numbers, Y_c and Y_s were known, we could compute the fundamental frequency phasor. With samples taken at an interval of T seconds,

$$y_n = y(nT) = Y_c \cos n\theta + Y_s \sin n\theta + e(nT) \quad (5.20)$$

where $\theta = \omega_0 T$ is the sampling angle. If signals other than the fundamental frequency signal were present, it would be useful to include them in a formulation similar to Eq. (5.19) so that they would be included in $e(t)$. If, for example, the second harmonic were included, Eq. (5.19) could be modified to

$$y_n = Y_{1c} \cos n\theta + Y_{1s} \sin n\theta + Y_{2c} \cos 2n\theta + Y_{2s} \sin 2n\theta + e(nT) \quad (5.21)$$

It is clear that more samples are needed to estimate the parameters as more terms are included. Equation (5.21) can be generalized to include any number of harmonics (the number is limited by the sampling rate), the exponential offset in a current, or any known signal that is suspected to be included in the post-fault waveform. No matter how detailed the formulation, $e(t)$ will include unpredictable contributions from:

- The transducers (CTs and PTs)
- Fault arc
- Traveling wave effects
- A/D converters

- The exponential offset in the current
- The transient response of the antialiasing filters
- The power system itself

The current offset is not an error signal for some algorithms and is removed separately for some others. The power system generated signals are transients depending on fault location, the fault incidence angle, and the structure of the power system. The power system transients are low enough in frequency to be present after the antialiasing filter.

We can write a general expression as

$$y_n = \sum_{k=1}^K s_k(nT)Y_k + e_n \quad (5.22)$$

If y represents a vector of N samples, and Y a vector of K unknown coefficients, then there are N equations in K unknowns in the form

$$y = SY + e \quad (5.23)$$

The matrix S is made up of samples of the signals s_k .

$$S = \begin{bmatrix} s_1(T) & s_2(T) & \cdots & s_K(T) \\ s_1(2T) & s_2(2T) & \cdots & s_K(2T) \\ \vdots & \vdots & \ddots & \vdots \\ s_1(NT) & s_2(NT) & \cdots & s_K(NT) \end{bmatrix} \quad (5.24)$$

The presence of the error e and the fact that the number of equations is larger than the number of unknowns ($N > K$) makes it necessary to estimate Y .

5.6.2 Least Squares Fitting

One criterion for choosing the estimate $\hat{Y}$ is to minimize the scalar formed as the sum of the squares of the error term in Eq. (5.23), viz.

$$e^T e = (y - SY)^T(y - SY) = \sum_{n=1}^N e_n^2 \quad (5.25)$$

It can be shown that the minimum least squared error [the minimum value of Eq. (5.25)] occurs when

$$\hat{Y} = (S^T S)^{-1} S^T y = By \quad (5.26)$$

where $B = (S^T S)^{-1} S^T$. The calculations involving the matrix S can be performed off-line to create an “algorithm,” i.e., an estimate of each of the K parameters is obtained by multiplying the N samples by a set of stored numbers. The rows of Eq. (5.26) can represent a number of different algorithms depending on the choice of the signals $s_k(nT)$ and the interval over which the samples are taken.

5.6.3 DFT

The simplest form of Eq. (5.26) is when the matrix $S^T S$ is diagonal. Using a signal alphabet of cosines and sines of the first N harmonics of the fundamental frequency over a window of one cycle of the fundamental frequency, the familiar Discrete Fourier Transform (DFT) is produced. With

$$\begin{aligned}
s_1(t) &= \cos(\omega_0 t) \\
s_2(t) &= \sin(\omega_0 t) \\
s_3(t) &= \cos(2\omega_0 t) \\
s_4(t) &= \sin(2\omega_0 t) \\
&\vdots \\
s_{N-1}(t) &= \cos(N\omega_0 t/2) \\
s_N(t) &= \sin(N\omega_0 t/2)
\end{aligned} \tag{5.27}$$

The estimates are given by :

$$\begin{aligned}
\hat{Y}_{Cp} &= \frac{2}{N} \sum_{n=0}^{N-1} y_n \cos(pn\theta) \\
\hat{Y}_{Sp} &= \frac{2}{N} \sum_{n=0}^{N-1} y_n \sin(pn\theta)
\end{aligned} \tag{5.28}$$

Note that the harmonics are also estimated by Eq. (5.28). Harmonics have little role in line relaying but are important in transformer protection. It can be seen that the fundamental frequency phasor can be obtained as

$$Y = \frac{2}{N\sqrt{2}} (Y_{C1} - jY_{S1}) \tag{5.29}$$

The normalizing factor in Eq. (5.29) is omitted if the ratio of phasors for voltage and current are used to form impedance.

5.6.4 Differential Equations

Another kind of algorithm is based on estimating the values of parameters of a physical model of the system. In line protection, the physical model is a series R-L circuit that represents the faulted line. A similar approach in transformer protection uses the magnetic flux circuit with associated inductance and resistance as the model. A differential equation is written for the system in both cases.

5.6.4.1 Line Protection Algorithms

The series R-L circuit of Fig. 5.6 is the model of a faulted line. The offset in the current is produced by the circuit model and hence will not be an error signal.

$$v(t) = Ri(t) + L \frac{di(t)}{dt} \tag{5.30}$$

Looking at the samples at $k, k + 1, k + 2$

$$\int_{t_0}^{t_1} v(t)dt = R \int_{t_0}^{t_1} i(t)dt + L(i(t_1) - i(t_0)) \tag{5.31}$$

$$\int_{t_1}^{t_2} v(t)dt = R \int_{t_1}^{t_2} i(t)dt + L(i(t_2) - i(t_1)) \tag{5.32}$$

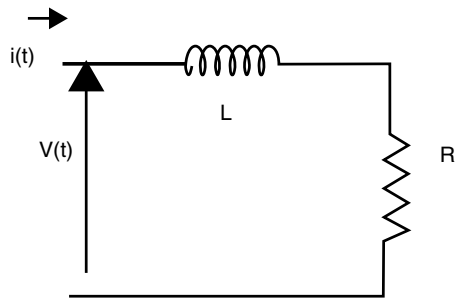


FIGURE 5.6 Model of a faulted line.

Using trapezoidal integration to evaluate the integrals (assuming t is small)

$$\int_{t_1}^{t_2} v(t)dt = R \int_{t_1}^{t_2} i(t)dt + L(i(t_2) - i(t_1)) \quad (5.33)$$

$$\int_{t_1}^{t_2} v(t)dt = R \int_{t_1}^{t_2} i(t)dt + L(i(t_2) - i(t_1)) \quad (5.34)$$

R and L are given by

$$R = \left[\frac{(v_{k+1} + v_k)(i_{k+2} - i_{k+1}) - (v_{k+2} + v_{k+1})(i_{k+1} - i_k)}{2(i_k i_{k+2} - i_{k+1}^2)} \right] \quad (5.35)$$

$$L = \frac{T}{2} \left[\frac{(v_{k+2} + v_{k+1})(i_{k+1} + i_k) - (v_{k+1} + v_k)(i_{k+2} + i_{k+1})}{2(i_k i_{k+2} - i_{k+1}^2)} \right] \quad (5.36)$$

It should be noted that the sample values occur in both numerator and denominator of Eqs. (5.35) and (5.36). The denominator is not constant but varies in time with local minima at points where both the current and the derivative of the current are small. For a pure sinusoidal current, the current and its derivative are never both small but when an offset is included there is a possibility of both being small once per period.

Error signals for this algorithm include terms that do not satisfy the differential equation such as the currents in the shunt elements in the line model required by long lines. In intervals where the denominator is small, errors in the numerator of Eqs. (5.35) and (5.36) are amplified. The resulting estimates can be quite poor. It is also difficult to make the window longer than three samples. The complexity of solving such equations for a larger number of samples suggests that the short window results be post processed. Simple averaging of the short-window estimates is inappropriate, however.

A counting scheme was used in which the counter was advanced if the estimated R and L were in the zone and the counter was decreased if the estimates lay outside the zone (Chen and Breingan, 1979). By requiring the counter to reach some threshold before tripping, secure operation can be assured with a cost of some delay. For example, if the threshold were set at six with a sampling rate of 16 times a cycle, the fastest trip decision would take a half cycle. Each “bad” estimate would delay the decision by two additional samples. The actual time for a relaying decision is variable and depends on the exact data.

The use of a median filter is an alternate to the counting scheme (Akke and Thorp, 1997). The median operation ranks the input values according to their amplitude and selects the middle value as the output. Median filters have an odd number of inputs. A length five median filter has an input-output relation between input $x[n]$ and output $y[n]$ given by

$$y[n] = \text{median}\{x[n-2], x[n-1], x[n], x[n+1], x[n+2]\} \quad (5.37)$$

Median filters of length five, seven, and nine have been applied to the output of the short window differential equation algorithm (Akke and Thorp, 1997). The median filter preserves the essential features of the input while removing isolated noise spikes. The filter length rather than the counter scheme, fixes the time required for a relaying decision.

5.6.4.2 Transformer Protection Algorithms

Virtually all algorithms for the protection of power transformers use the principle of percentage differential protection. The difference between algorithms lies in how the algorithm restrains the differential trip for conditions of overexcitation and inrush. Algorithms based on harmonic restraint, which

parallel existing analog protection, compute the second and fifth harmonics using Eq. (5.10) (Thorp and Phadke, 1982). These algorithms use current measurements only and cannot be faster than one cycle because of the need to compute the second harmonic. The harmonic calculation provides for secure operation since the transient event produces harmonic content which delays relay operation for about a cycle.

In an integrated substation with other microprocessor relays, it is possible to consider transformer protection algorithms that use voltage information. Shared voltage samples could be a result of multiple protection modules connected in a LAN in the substation. The magnitude of the voltage itself can be used as a restraint in a digital version of a “tripping suppressor” (Harder and Marter, 1948). A physical model similar to the differential equation model for a faulted line can be constructed using the flux in the transformer. The differential equation describing the terminal voltage, $v(t)$, the winding current, $i(t)$, and the flux linkage $\Lambda(t)$ is:

$$v(t) - L \frac{di(t)}{dt} = \frac{d\Lambda(t)}{dt} \quad (5.38)$$

where L is the leakage inductance of the winding.

Using trapezoidal integration for the integral in Eq. (5.38)

$$\int_{t_1}^{t_2} v(t) dt - L[i(t_2) - i(t_1)] = \Lambda(t_2) - \Lambda(t_1) \quad (5.39)$$

gives

$$\Lambda(t_2) - \Lambda(t_1) = \frac{T}{2} [v(t_2) + v(t_1)] - L[i(t_2) - i(t_1)] \quad (5.40)$$

or

$$\Lambda_{k+1} = \Lambda_k + \frac{T}{2} [v_{k+1} + v_k] - L[i_{k+1} - i_k] \quad (5.41)$$

Since the initial flux Λ_0 in Eq. (5.41) cannot be known without separate sensing, the slope of the flux current curve is used

$$\left(\frac{d\Lambda}{di} \right)_k = \frac{T}{2} \left[\frac{v_k + v_{k-1}}{i_k - i_{k-1}} \right] - L \quad (5.42)$$

The slope of the flux current characteristic shown in Fig. 5.7 is different depending on whether there is a fault or not. The algorithm must then be able to differentiate between inrush (the slope alternates between large and small values) and a fault (the slope is always small). The counting scheme used for the differential equation algorithm for line protection can be adapted to this application. The counter increases if the slope is less than a threshold and the differential current indicates trip, and the counter decreases if the slope is greater than the threshold or the differential does not indicate trip.

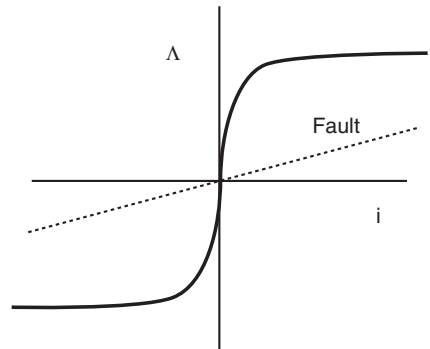


FIGURE 5.7 The flux-current characteristic compared to fault conditions.

5.6.5 Kalman Filters

The Kalman filter provides a solution to the estimation problem in the context of an evolution of the parameters to be estimated according to a state equation. It has been used extensively in estimation problems for dynamic systems. Its use in relaying is motivated by the filter's ability to handle measurements that change in time. To model the problem so that a Kalman filter may be used, it is necessary to write a state equation for the parameters to be estimated in the form

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{w}_k \quad (5.43)$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \quad (5.44)$$

where Eq. (5.43) (the state equation) represents the evolution of the parameters in time and Eq. (5.44) represents the measurements. The terms $\mathbf{w}_k$ and $\mathbf{v}_k$ are discrete time random processes representing state noise, i.e., random inputs in the evolution of the parameters, and measurement errors, respectively. Typically $\mathbf{w}_k$ and $\mathbf{v}_k$ are assumed to be independent of each other and uncorrelated from sample to sample. If $\mathbf{w}_k$ and $\mathbf{v}_k$ have zero means, then it is common to assume that

$$\begin{aligned} E\{\mathbf{w}_k \mathbf{w}_j^T\} &= \mathbf{Q}_k : k = j \\ &= 0; \quad k \neq j \end{aligned} \quad (5.45)$$

The matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$ are the covariance matrices of the random processes and are allowed to change as k changes. The matrix Φ_k in Eq. (5.43) is the state transition matrix. If we imagine sampling a pure sinusoid of the form

$$y(t) = Y_c \cos(\omega t) + Y_s \sin(\omega t) \quad (5.46)$$

at equal intervals corresponding to $\omega\Delta\tau = \Psi$, then the state would be

$$\mathbf{x}_k = \begin{bmatrix} Y_C \\ Y_S \end{bmatrix} \quad (5.47)$$

and the state transition matrix

$$\Phi_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (5.48)$$

In this case, $\mathbf{H}_k$, the measurement matrix, would be

$$\mathbf{H}_k = [\cos(k\Psi) \sin(k\Psi)] \quad (5.49)$$

Simulations of a 345 kV line connecting a generator and a load (Gurgis and Brown, 1981) led to the conclusion that the covariance of the noise in the voltage and current decayed in time. If the time constant of the decay is comparable to the decision time of the relay, then the Kalman filter formulation is

$$\mathbf{x} = \begin{bmatrix} Y_C \\ Y_S \\ Y_0 \end{bmatrix} \quad \Phi_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{-\beta t} \end{bmatrix}$$

appropriate for the estimation problem. The voltage was modeled as in Eqs. (5.48) and (5.49). The current was modeled with three states to account for the exponential offset.

and

$$H_k = [\cos(k\Psi) \sin(k\Psi) 1] \quad (5.50)$$

The measurement covariance matrix was

$$R_k = K e^{-k\Delta t/T} \quad (5.51)$$

with T chosen as half the line time constant and different K s for voltage and current. The Kalman filter estimates phasors for voltage and current as the DFT algorithms. The filter must be started and terminated using some other software. After the calculations begin, the data window continues to grow until the process is halted. This is different from fixed data windows such as a one cycle Fourier calculation. The growing data window has some advantages, but has the limitation that if started incorrectly, it has a hard time recovering if a fault occurs after the calculations have been initiated.

The Kalman filter assumes an initial statistical description of the state x , and recursively updates the estimate of state. The initial assumption about the state is that it is a random vector independent of the processes w_k and v_k and with a known mean and covariance matrix, P_0 . The recursive calculation involves computing a gain matrix K_k . The estimate is given by

$$\hat{x}_{k+1} = \Phi_k \hat{x}_k + K_{k+1} [z_{k+1} - H_{k+1} \hat{x}_k] \quad (5.52)$$

The first term in Eq. (5.52) is an update of the old estimate by the state transition matrix while the second is the gain matrix K_{k+1} multiplying the observation residual. The bracketed term in Eq. (5.52) is the difference between the actual measurement, z_k , and the predicted value of the measurement, i.e., the residual in predicting the measurement. The gain matrix can then be computed recursively. The amount of computation involved depends on the state vector dimension. For the linear problem described here, these calculations can be performed off-line. In the absence of the decaying measurement error, the Kalman filter offers little other than the growing data window. It has been shown that at multiples of a half cycle, the Kalman filter estimate for a constant error covariance is the same as that obtained from the DFT.

5.6.6 Wavelet Transforms

The Wavelet Transform is a signal processing tool that is replacing the Fourier Transform in many applications including data compression, sonar and radar, communications, and biomedical applications. In the signal processing community there is considerable overlap between wavelets and the area of filter banks. In applications in which it is used, the Wavelet Transform is viewed as an improvement over the Fourier Transform because it deals with time-frequency resolution in a different way. The Fourier Transform provides a decomposition of a time function into exponentials, $e^{j\omega t}$, which exist for all time. We should consider the signal that is processed with the DFT calculations in the previous sections as being extended periodically for all time. That is, the data window represents one period of a periodic signal. The sampling rate and the length of the data window determine the frequency resolution of the calculations. While these limitations are well understood and intuitive, they are serious limitations in some applications such as compression. The Wavelet Transform introduces an alternative to these limitations.

The Fourier Transform can be written

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (5.53)$$

The effect of a data window can be captured by imagining that the signal $x(t)$ is windowed before the Fourier Transform is computed. The function $h(t)$ represents the windowing function such as a one-cycle rectangle.

$$X(\omega, t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau)e^{-j\omega\tau} d\tau \quad (5.54)$$

The Wavelet Transform is written

$$X(s, t) = \int_{-\infty}^{\infty} x(\tau) \left[\frac{1}{\sqrt{s}} h\left(\frac{\tau - t}{s}\right) \right] d\tau \quad (5.55)$$

where s is a scale parameter and t is a time shift. The scale parameter is an alternative to the frequency parameter of the Fourier Transform. If $h(t)$ has Fourier Transform $H(\omega)$, then $h(t/s)$ has Fourier Transform $H(s\omega)$. Note that for a fixed $h(t)$ that large, s compresses the transform while small s spreads the transform in frequency. There are a few requirements on a signal $h(t)$ to be the “mother wavelet” (essentially that $h(t)$ have finite energy and be a bandpass signal). For example, $h(t)$ could be the output of a bandpass filter. It is also true that it is only necessary to know the Wavelet Transform at discrete values of s and t in order to be able to represent the signal. In particular

$$\begin{aligned} s &= 2^m, \quad t = n2^m \quad m = \dots, -2, 0, 1, 2, 3, \dots \\ n &= \dots, -2, 0, 1, 2, 3, \dots \end{aligned}$$

where lower values of m correspond to smaller values of s or higher frequencies.

If $x(t)$ is limited to a band B Hz, then it can be represented by samples at $T_s = 1/2B$ sec.

$$x(n) = x(nT_s)$$

Using a mother wavelet corresponding to an ideal bandpass filter illustrates a number of ideas. [Figure 5.8](#) shows the filters corresponding to $m = 0, 1, 2$, and 3 and [Fig. 5.9](#) shows the corresponding time functions. Since $x(t)$ has no frequencies above B Hz, only positive values of m are necessary. The structure of the process can be seen in [Fig. 5.10](#). The boxes labeled LPF_R and HPF_R are low and high pass filters with cutoff frequencies of R Hz. The circle with the down arrow and a 2 represents the process of taking every other sample. For example, on the first line the output of the bandpass filter only has a bandwidth of $B/2$ Hz and the samples at T_s sec can be decimated to samples at $2T_s$ sec.

Additional understanding of the compression process is possible if we take a signal made of eight numbers and let the low pass filter be the average of two consecutive samples $(x(n) + x(n + 1))/2$ and the high pass filter to be the difference $(x(n) - x(n + 1))/2$ (Gail and Nielsen, 1999). For example, with

$$x(n) = [-2 \ -28 \ -46 \ -44 \ -20 \ 12 \ 32 \ 30]$$

we get

$$\begin{aligned} h_1(k_1) &= [13 \ -1 \ -16 \ 1] \\ h_2(k_2) &= [7 \ -8.5] \\ h_3(k_3) &= [7.75] \\ l_3(k_3) &= [-0.75] \end{aligned}$$

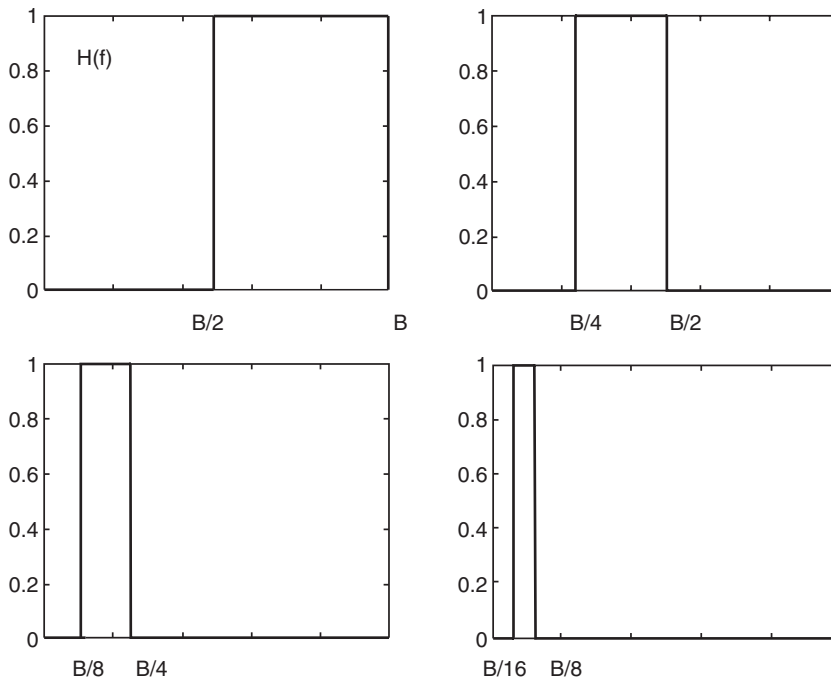


FIGURE 5.8 Ideal bandpass filters corresponding to $m = 0, 1, 2, 3$.

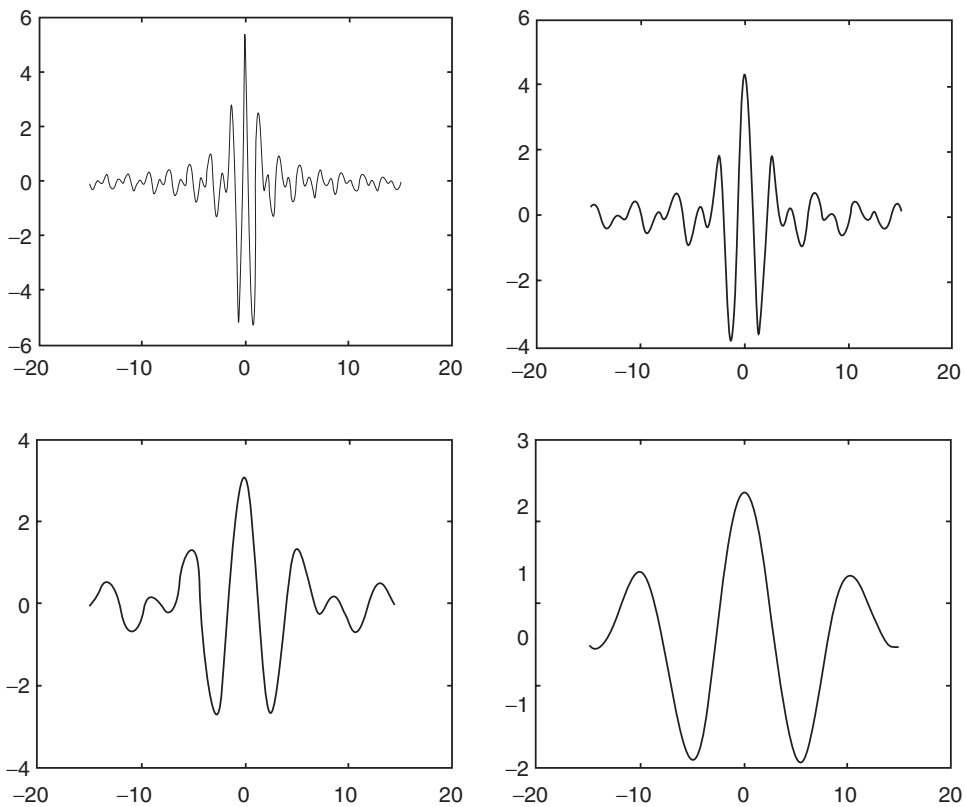


FIGURE 5.9 The impulse responses corresponding to the filters in Fig. 5.8.

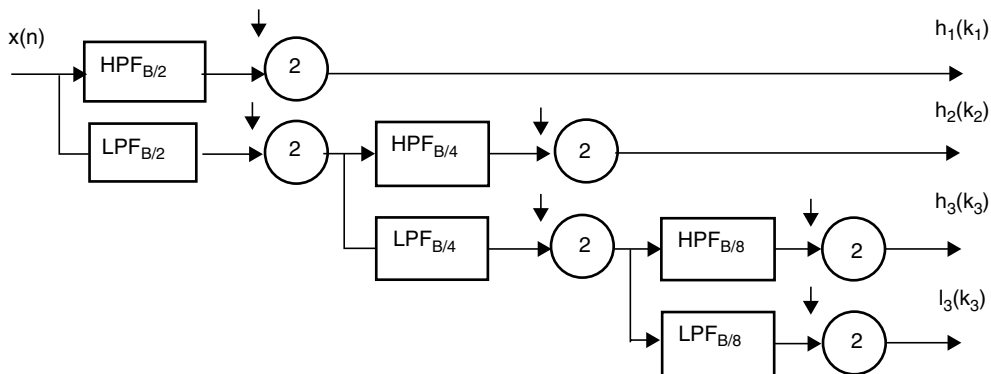


FIGURE 5.10 Cascade filter structure.

If we truncate to form

$$\begin{aligned} h_1(k_1) &= [16 \ 0 \ -16 \ 0] \\ h_2(k_2) &= [8 \ -8] \\ h_3(k_3) &= [8] \\ l_3(k_3) &= [0] \end{aligned}$$

and reconstruct the original sequence

$$\tilde{x}(n) = [0 \ -32 \ -48 \ -48 \ -24 \ 8 \ 32 \ 32]$$

The original and reconstructed compressed waveform is shown in Fig. 5.11. Wavelets have been applied to relaying for systems grounded through a Peterson coil where the form of the wavelet was chosen to fit unusual waveforms the Peterson coil produces (Chaari et al., 1996).

5.6.7 Neural Networks

Artificial Neural Networks (ANNs) had their beginning in the “perceptron,” which was designed to recognize patterns. The number of papers suggesting relay application have soared. The attraction is the

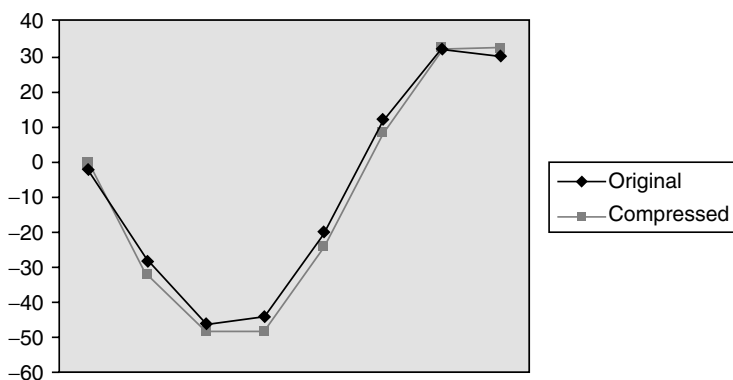


FIGURE 5.11 Original and compressed signals.

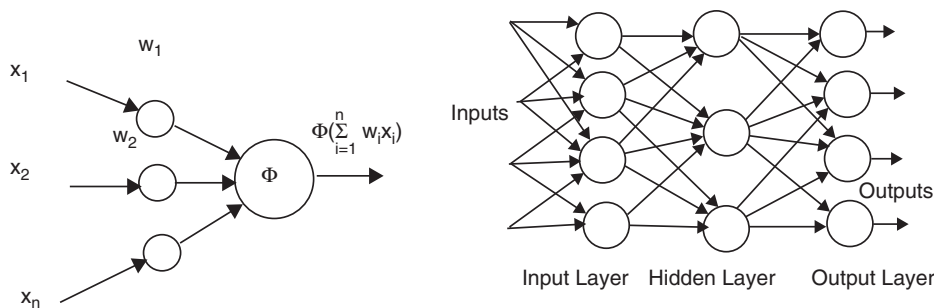


FIGURE 5.12 One neuron and a neural network.

use of ANNs as pattern recognition devices that can be trained with data to recognize faults, intrush, or other protection effects. The basic feed forward neural net is composed of layers of neurons as shown in Fig. 5.12.

The function Φ is either a threshold function or a saturating function such as a symmetric sigmoid function. The weights w_i are determined by training the network. The training process is the most difficult part of the ANN process. Typically, simulation data such as that obtained from EMTP is used to train the ANN. A set of cases to be executed must be identified along with a proposed structure for the net. The structure is described in terms of the number of inputs, neuron in layers, various layers, and outputs. An example might be a net with 12 inputs, and a 4, 3, 1 layer structure. There would be 4×12 plus 4×3 plus 3×1 or 63 weights to be determined. Clearly, a lot more than 60 training cases are needed to learn 63 weights. In addition, some cases not used for training are needed for testing. Software exists for the training process but judgment in determining the training sequences is vital. Once the weights are learned, the designer is frequently asked how the ANN will perform when some combination of inputs are presented to it. The ability to answer such questions is very much a function of the breadth of the training sequence.

The protective relaying application of ANNs include high-impedance fault detection (Eborn et al., 1990), transformer protection (Perez et al., 1994), fault classification (Dalstein and Kulicke, 1995), fault direction determination, adaptive reclosing (Aggarwal et al., 1994), and rotating machinery protection (Chow and Yee, 1991).

References

- Aggarwal, R.K., Johns, A.T., Song, Y.H., Dunn, R.W., and Fitton, D.S., Neural-network based adaptive single-pole autoreclosure technique for EHV transmission systems, *IEEE Proceedings—C*, 141, 155, 1994.
- Akke, M. and Thorp, J.S., Improved estimates from the differential equation algorithm by median post-filtering, *IEEE Sixth Int. Conf. on Development in Power System Protection*, Univ. of Nottingham, UK, March 1997.
- Chaari, O., Neunier, M., and Brouaye, F., Wavelets: A new tool for the resonant grounded power distribution system relaying, *IEEE Trans. on Power Delivery*, 11, 1301, July 1996.
- Chen, M.M. and Breingan, W.D., Field experience with a digital system with transmission line protection, *IEEE Trans. on Power Appar. and Syst.*, 98, 1796, Sep./Oct. 1979.
- Chow, M. and Yee, S.O., Methodology for on-line incipient fault detection in single-phase squirrel-cage induction motors using artificial neural networks, *IEEE Trans. on Energy Conversion*, 6, 536, Sept. 1991.
- Dalstein, T. and Kulicke, B., Neural network approach to fault classification for high speed protective relaying, *IEEE Trans. on Power Delivery*, 10, 1002, Apr. 1995.

- Eborn, S., Lubkeman, D.L., and White, M., A neural network approach to the detection of incipient faults on power distribution feeders, *IEEE Trans. on Power Delivery*, 5, 905, Apr. 1990.
- Gail, A.W. and Nielsen, O.M., Wavelet analysis for power system transients, *IEEE Computer Applications in Power*, 12, 16, January 1999.
- Gilcrest, G.B., Rockefeller, G.D., and Udren, E.A., High-speed distance relaying using a digital computer, Part I: System description, *IEEE Trans.*, 91, 1235, May/June 1972.
- Girgis, A.A. and Brown, R.G., Application of Kalman filtering in computer relaying, *IEEE Trans. on Power Appar. and Syst.*, 100, 3387, July 1981.
- Harder, E.L. and Marter, W.E., Principles and practices of relaying in the United States, *AIEE Transactions*, 67, Part II, 1005, 1948.
- Mann, B.J. and Morrison, I.F., Relaying a three-phase transmission line with a digital computer, *IEEE Trans. on Power Appar. and Syst.*, 90, 742, Mar./April 1971.
- Perez, L.G., Flechsiz, A.J., Meador, J.L., and Obradovic, A., Training an artificial neural network to discriminate between magnetizing inrush and internal faults, *IEEE Trans. on Power Delivery*, 9, 434, Jan. 1994.
- Poncelet, R., The use of digital computers for network protection, *CIGRE*, 32–98, Aug. 1972.
- Rockefeller, G.D., Fault protection with a digital computer, *IEEE Trans.*, PAS-88, 438, Apr. 1969.
- Rockefeller, G.D. and Udren, E.A., High-speed distance relaying using a digital computer, Part II. Test results, *IEEE Trans.*, 91, 1244, May/June 1972.
- Sachdev, M.S. (Coordinator), Advancements in microprocessor based protection and communication, *IEEE Tutorial Course Text Publication*, 97TP120–0, 1997.
- Tamronglak, S., Horowitz, S.H., Phadke, A.G., and Thorp, J.S., Anatomy of power system blackouts: Preventive relaying strategies, *IEEE Trans. on Power Delivery*, 11, 708, Apr. 1996.
- Thorp, J.S. and Phadke, A.G., A microprocessor based voltage-restraint three-phase transformer differential relay, *Proc. South Eastern Symp. on Systems Theory*, 312, April 1982.